# GETLOGIN

Results of getlogin() should not be trusted

Sean Barnum, Cigital, Inc. [vita[1]]

2007-03-23

## Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 4079 bytes

<table>
<tr><td><b>Attack Category</b></td><td>• Identity Spoofing</td></tr>
<tr><td><b>Vulnerability Category</b></td><td>• Privilege escalation problem<br>• Access Control</td></tr>
<tr><td><b>Software Context</b></td><td>• Authorization</td></tr>
<tr><td><b>Location</b></td><td></td></tr>
<tr><td><b>Description</b></td><td>The results of getlogin() should not be trusted.<br>The getlogin() function returns a pointer to a string that contains the name of the user associated with the calling process. The function is not reentrant, meaning that if it is called from another process, the contents are not locked out and the value of the string can be changed by another process. This makes it very risky to use because the username can be changed by other processes, so the results of the function cannot be trusted.<br>Also, according to the Linux man page: "Unfortunately, it is often rather easy to fool getlogin(). Sometimes it does not work at all, because some program messed up the utmp file. Often, it gives only the first 8 characters of the login name. The user currently logged in on the controlling tty of our program need not be the user who started it. Avoid getlogin() for security-related purposes."<br>Guidance: Using names for security purposes is not advised. Names are easy to forge and can have overlapping user IDs, potentially causing confusion or impersonation.</td></tr>
<tr><td><b>APIs</b></td><td><table><tr><th>Function Name</th><th>Comments</th></tr><tr><td>getlogin</td><td></td></tr></table></td></tr>
<tr><td><b>Method of Attack</b></td><td></td></tr>
<tr><td><b>Exception Criteria</b></td><td></td></tr>
<tr><td><b>Solutions</b></td><td><table><tr><th>Solution Applicability</th><th>Solution Description</th><th>Solution Efficacy</th></tr></table></td></tr>
</table>

[1] http://buildsecurityin.us-cert.gov/bsi-rules/35-BSI.html (Barnum, Sean)

<table>
<tr><td></td><td><table><tr><td>When login user name is needed.</td><td>Never trust the results of the function getlogin(). Use getlogin_r() instead, which is reentrant, meaning that other processes are locked out from changing the username.</td><td>Somewhat effective. Eliminates risk of buffer being overwritten, but other risks may persist.</td></tr></table></td></tr>
<tr><td><b>Signature Details</b></td><td>char *getlogin(void);</td></tr>
<tr><td><b>Examples of Incorrect Code</b></td><td><pre>/* The following is not
trustworthy */

char *userName = getlogin();</pre></td></tr>
<tr><td><b>Examples of Corrected Code</b></td><td><pre>/* The following is more
trustworthy, but still has
possible issues */

const int userNameSize =
L_cuserid; // Standard macro
giving size of user names
char userName[userNameSize];
if (getlogin_r(userName,
userNameSize)) { handleError(); }</pre></td></tr>
<tr><td><b>Source Reference</b></td><td>• Rough Auditing Tool for Security (RATS)[2]</td></tr>
<tr><td><b>Recommended Resource</b></td><td>• Linux man page for getlogin(), getlogin_r()[3]</td></tr>
<tr><td><b>Discriminant Set</b></td><td><table><tr><td><b>Operating System</b></td><td>• Windows</td></tr><tr><td><b>Languages</b></td><td>• C<br>• C++</td></tr></table></td></tr>
</table>

# Cigital, Inc. Copyright

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.

[1] mailto:copyright@cigital.com